

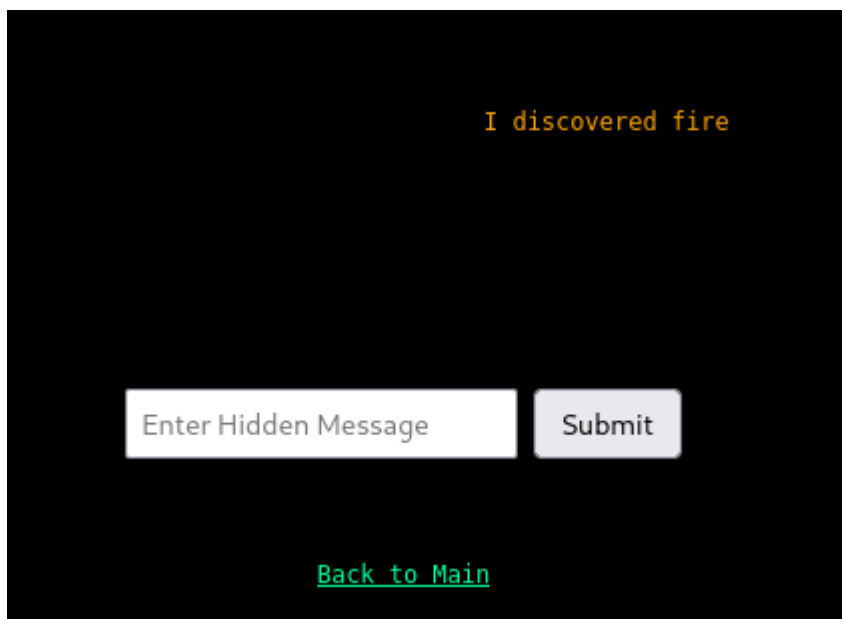
ESG TimeMachine Walkthrough

01. Prehistory

힌트를 보면 무엇인가를 찾아야 함을 알 수 있다.

Hint: Discover the fire hidden in the darkness.

마우스 커서를 사용하거나 개발자 도구, 페이지 소스를 통해 숨겨진 문장을 찾을 수 있다.



I discovered fire 입력하면 클리어

02. Ancient Hieroglyphs

힌트를 보면 주소창에 php?hint=hint를 입력해야 함을 알 수 있다.

hint=hint

php?hint=hint로 들어가면 각 상형문자에 대응하는 알파벳과 숫자를 알 수 있다.



각각 대응해보면 BRING THE NEXT STONE 이 나온다. 이를 입력하면 클리어

03. Ancient Greek Forum

검색창에 주어진 예시처럼 Socrates를 입력하면 다음과 같이 나온다.

| <input type="text" value="e.g., Socrates"/> <input type="button" value="Search"/> | |
|---|--|
| Philosopher | Quote |
| Socrates | The only true wisdom is in knowing you know nothing. |

sql injection을 시도하기 위해 'or 1=1#를 입력하면 데이터가 모두 나온다.

| Philosopher | Quote |
|-------------|--|
| Socrates | The only true wisdom is in knowing you know nothing. |
| Plato | Wise men speak because they have something to say; fools because they have to say something. |
| Aristotle | It is the mark of an educated mind to be able to entertain a thought without accepting it. |
| Descartes | Je pense, donc je suis. |

뭔가 다른 것을 적으라고 되어있으므로 Je pense, donc je suis. 입력하면 클리어

04. Middle Age : Witch Hunt

미션을 보면 가입을 해서 admin으로 위장하는 것이 목적이다.

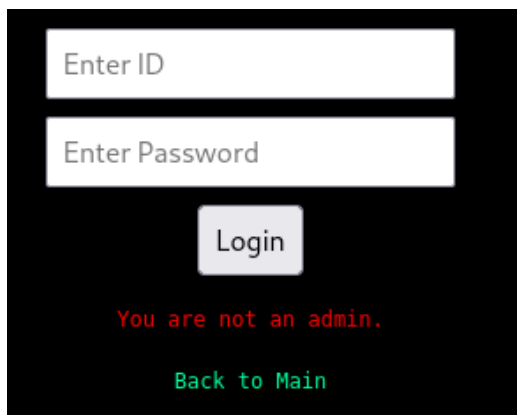
mission: Join the witch gathering and disguise admin.

test /1234 로 회원가입 해보자.



A screenshot of a web form titled "Join the Witch Gathering". It features two input fields: the first contains the text "test" and the second contains four dots, indicating a password field. Below the fields is a "Join" button. At the bottom of the form, there is a link labeled "Back to Main" in green text.

test /1234 로 로그인하면 admin이 아니라고 뜬다.



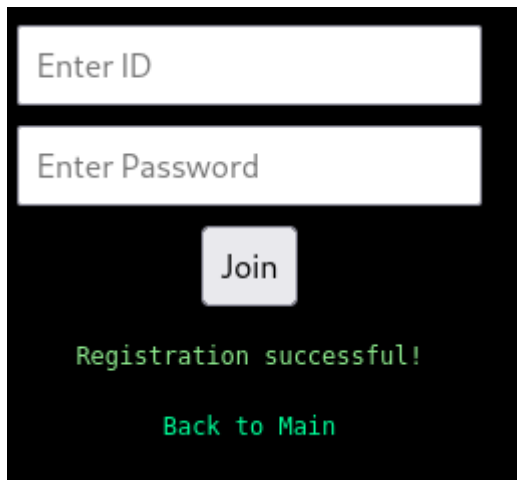
A screenshot of a login form with two input fields labeled "Enter ID" and "Enter Password". Below the fields is a "Login" button. A red error message "You are not an admin." is displayed below the button. At the bottom, there is a link labeled "Back to Main" in green text.

admin / 1234 로 회원가입 해보자. admin으로는 로그인이 안된다.



A screenshot of a join form with two input fields labeled "Enter ID" and "Enter Password". Below the fields is a "Join" button. A red error message "This ID cannot be used." is displayed below the button. At the bottom, there is a link labeled "Back to Main" in green text.

admin123 / 1234 로 회원가입 해본다. 회원가입이 성공했다.

A screenshot of a registration success screen. It features a black background with two white input fields at the top, labeled 'Enter ID' and 'Enter Password'. Below these is a grey 'Join' button. Underneath the button, the text 'Registration successful!' is displayed in green. At the bottom, there is a green link that says 'Back to Main'.

admin123 / 1234 로 로그인 하면 성공.

admin+@ 로 로그인 시도하면 성공한다.

05. Scientific Revolution

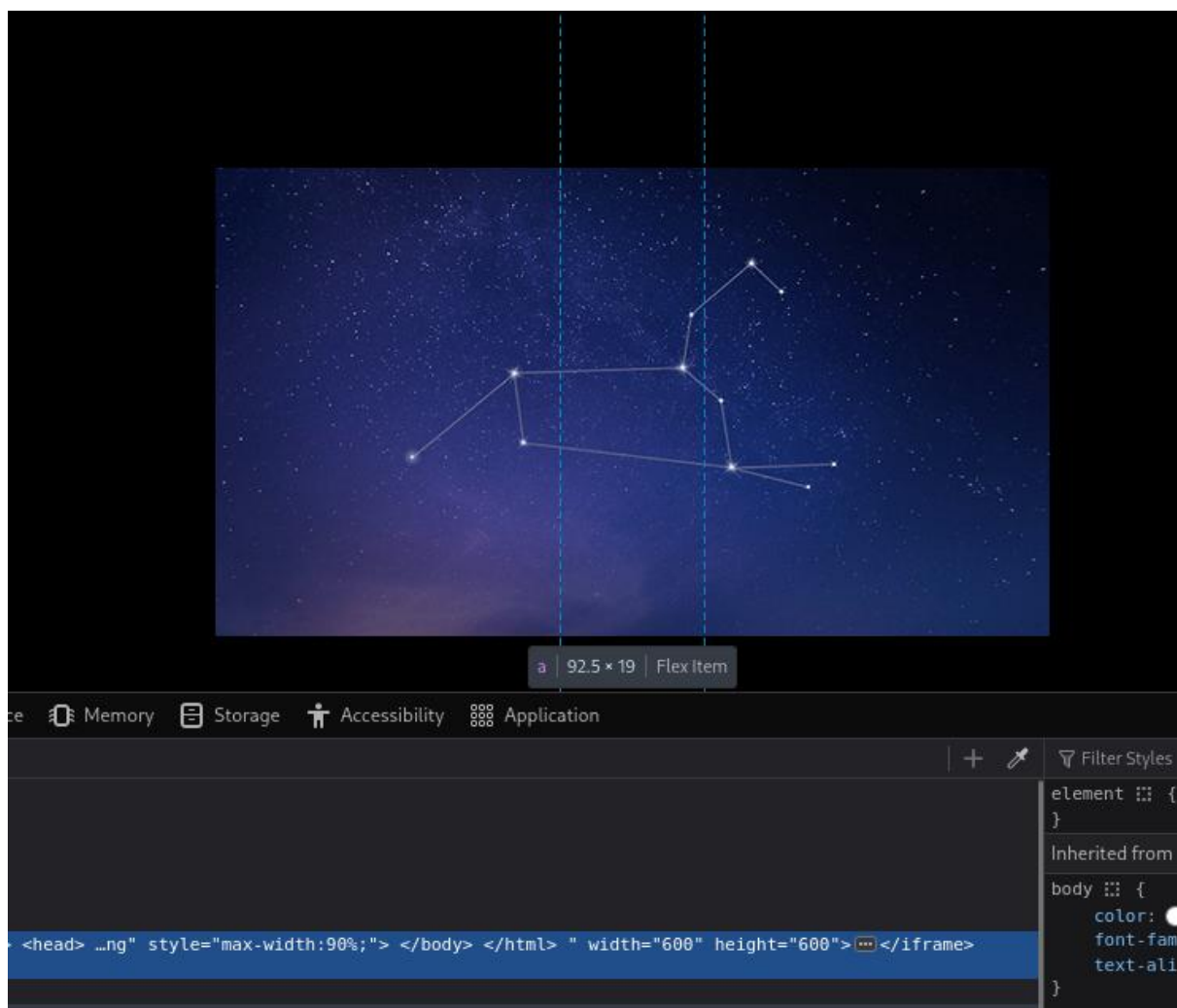
telescope을 사용하라고 되어있다.

To prove our claim, we must use a telescope.

개발자모드를 확인하면

```
<iframe id="telescope" width="0" height="0" frameborder="0" srcdoc="
```

width와 height가 0이므로 숫자를 늘려보면 별자리 이미지가 나온다.



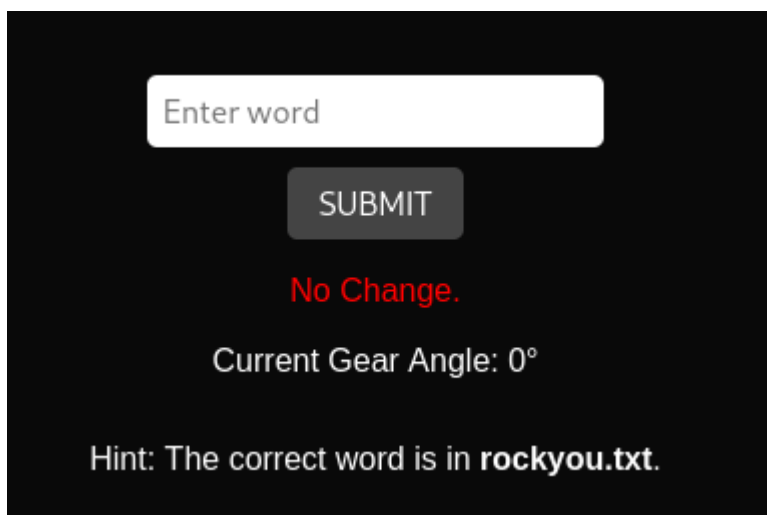
답에서 별자리를 입력하라고 되어 있으므로 사자자리인 leo를 입력하면 성공.

06. Industrial Revolution

톱니바퀴를 5바퀴 돌려야 한다고 나와있다.



입력칸에 아무 단어나 입력했더니 No change가 뜬다. 힌트로 rockyou.txt에 들어갈 단어가 있다는 것을 알았으니 hydra를 이용해서 보겠다.

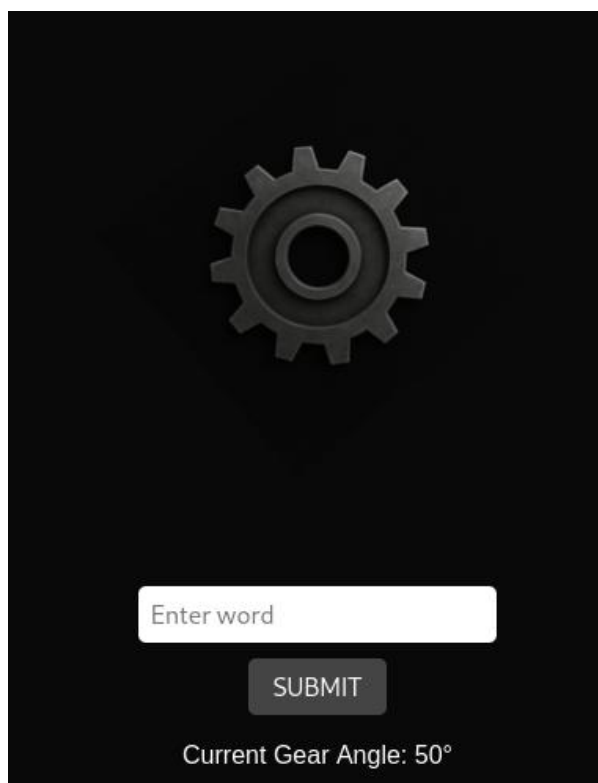


Hydra를 통해 mamaguebo라는 단어를 얻었다.

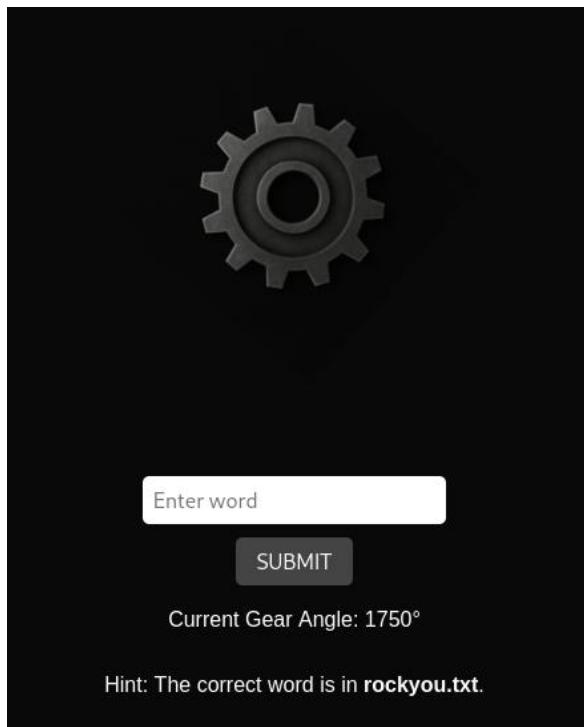
```
(root@kali-lgh)~[~]
# hydra -l user -P /usr/share/wordlists/rockyou.txt 192.168.5.166 http-post-form "/problems/timemachine/06.php:flag=^PASS^:No Change." -t 64
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-05 22:57:24
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 64 tasks per 1 server, overall 64 tasks, 14344399 login tries (l:1/p:14344399), ~224132 tries per task
[DATA] attacking http-post-form://192.168.5.166:80/problems/timemachine/06.php:flag=^PASS^:No Change.
[STATUS] 11896.00 tries/min, 11896 tries in 00:01h, 14332503 to do in 20:05h, 64 active
[STATUS] 11960.67 tries/min, 35882 tries in 00:03h, 14308517 to do in 19:57h, 64 active
[STATUS] 11959.86 tries/min, 83719 tries in 00:07h, 14260680 to do in 19:53h, 64 active
[80][http-post-form] host: 192.168.5.166 login: user password: mamaguebo
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-05 23:05:00
```

이를 대입하면 바퀴가 50도 움직인다.



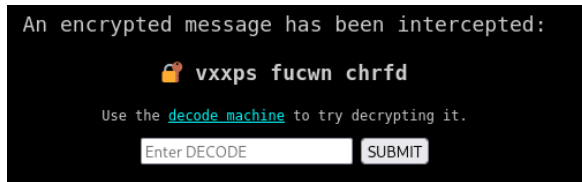
5바퀴는 1800도이므로 35번 더 움직이게 하면 된다. 일일이 복사 붙여넣기 해서 돌려도 되지만 burp suite의 repeater를 이용해서 돌리면 수월하다.



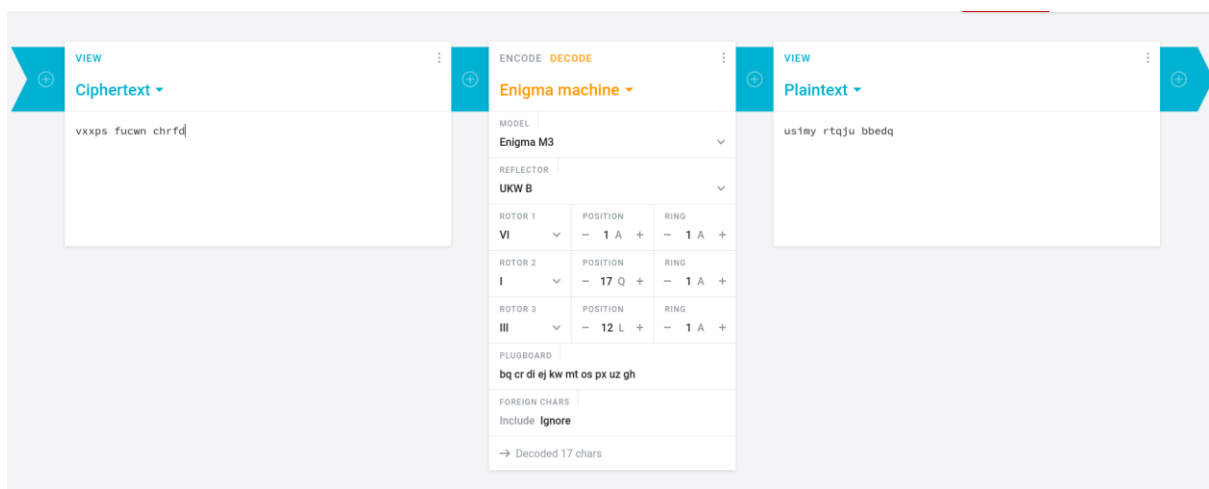
마지막 한번을 더 입력하면 클리어.

07. Enigma

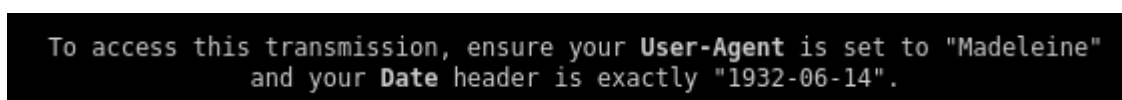
암호화된 메시지를 해독해야 한다. decode machine 링크를 이용해서 디코딩하자.



decode machine을 누르면 enigma를 디코딩할 수 있는 사이트가 나오는데 좌측 view에 암호문을 적는다.



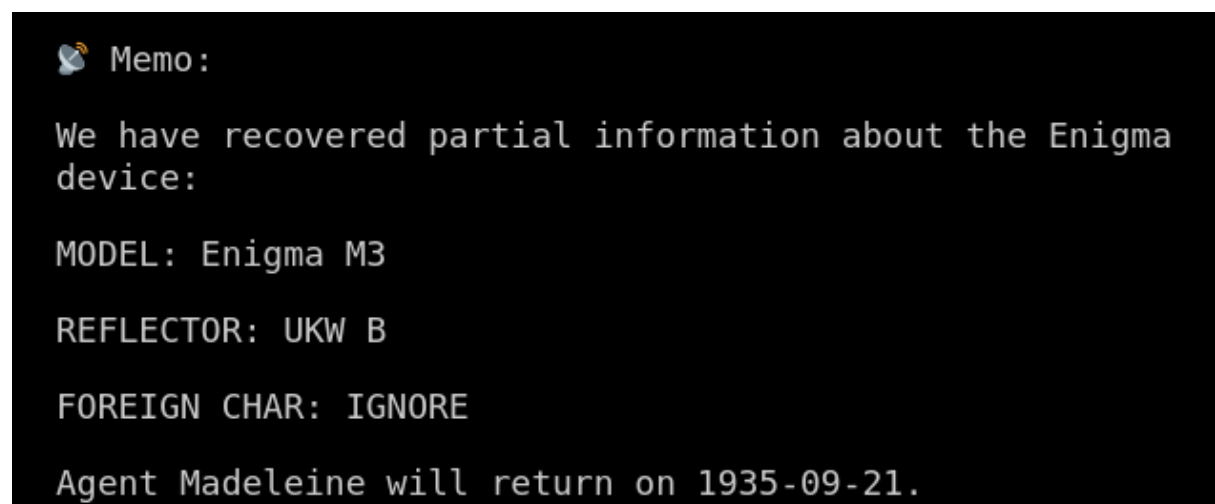
아래 문장에 따라 http 헤더에서 User-Agent는 Madeleine을 입력하고 Date에는 1932-06-14를 입력해보자.



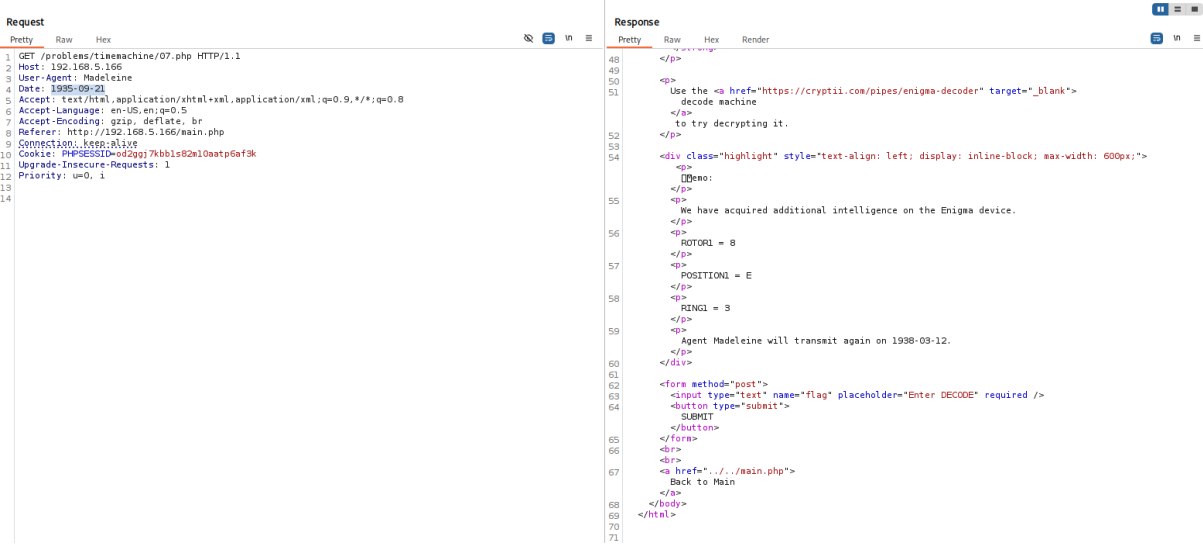
Burp suite를 이용하여 User-Agent와 Date값을 입력한다. Repeater를 이용하면 바로 입력하여 확인할 수 있다.

```
1 GET /problems/timemachine/07.php HTTP/1.1
2 Host: 192.168.5.166
3 User-Agent: Madeleine
4 Date: 1992-06-14
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: http://192.168.5.166/main.php
9 Cookie: PHPSESSID=ed2ggj7kbbis82nl0atp6ef3k
10
11 Upgrade-Insecure-Requests: 1
12 Priority: u=0, i
13
14
48 </p>
49
50 <p>
51     Use the <a href="https://cryptii.com/pipes/enigma-decoder" target="_blank">
52         decode machine
53     </a>
54     to try decrypting it.
55 </p>
56 <div class="highlight" style="text-align: left; display: inline-block; max-width: 600px;">
57     <p>
58         INFO:
59         </p>
60         We have recovered partial information about the Enigma device:
61         </p>
62         MODEL: Enigma M3
63         </p>
64         REFLECTOR: UKW B
65         </p>
66         FOREIGN CHAR: IGNORE
67         </p>
68         Agent Madeleine will return on 1935-09-21.
69     </div>
70 </p>
71 <form method="post">
72     <input type="text" name="flag" placeholder="Enter DECODE" required />
73     <button type="submit">
74         SUBMIT
75     </button>
76 </form>
77 <br>
78 <a href="...">
79     Back to Main
80 </a>
81 </body>
82 </html>
```

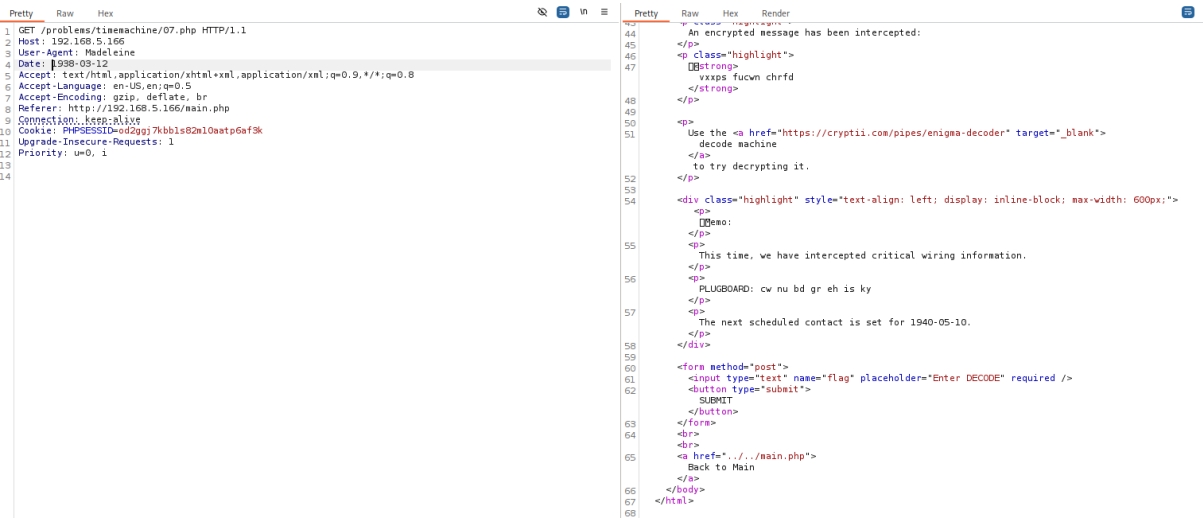
웹에는 다음과 같이 나온다. 힌트에 따라 decode 사이트에서 수정한다.



Date 값을 1935-09-21로 수정한다. 힌트에 따라 Rotor1을 8, Position1을 E, Ring1을 3으로 수정한다.



Date값을 1938-03-12로 변경한다. Plugboard 값을 cw nu bd gr eh is ky로 변경한다.



다음으로 Date값을 1940-05-10로 변경한다. Rotor2를 5, Position2를 F, Ring2를 12로 변경한다.

| Request | | | | Response | | | |
|---|-----|-----|--|---|-----|-----|--------|
| Pretty | Raw | Hex | | Pretty | Raw | Hex | Render |
| 1 GET /problems/timemachine/07.php HTTP/1.1 | | | | 48 </p> | | | |
| 2 Host: 192.168.5.166 | | | | 49 <p> | | | |
| 3 User-Agent: Wadeline | | | | 50 <p>Use the | | | |
| 4 Date: 1940-05-10 | | | | 51 decode machine | | | |
| 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 | | | | | | | |
| 6 Accept-Language: en-US,en;q=0.5 | | | | to try decrypting it. | | | |
| 7 Accept-Encoding: gzip, deflate, br | | | | </p> | | | |
| 8 Referer: http://192.168.5.166/main.php | | | | 52 </div> | | | |
| 9 Cookie: PHPSESSID=vd2ggj7kbb1s82n10atp6af3k | | | | 53 <div class="highlight" style="text-align: left; display: inline-block; max-width: 600px;"> | | | |
| 10 Upgrade-Insecure-Requests: 1 | | | | 54 <p> | | | |
| 11 Priority: u=0, i | | | | 55 <p>Demo: | | | |
| 12 | | | | 56 <p>We have obtained details about the second rotor of the Enigma machine. | | | |
| 13 | | | | 57 <p>ROTOR2 = 5 | | | |
| 14 | | | | 58 <p>POSITION2 = F | | | |
| | | | | 59 <p>RING2 = 12 | | | |
| | | | | 60 <p>The next message will arrive on 1941-11-03. We'll see you then. | | | |
| | | | | 61 </div> | | | |
| | | | | 62 <form method="post"> | | | |
| | | | | 63 <input type="text" name="flag" placeholder="Enter DECODE" required /> | | | |
| | | | | 64 <button type="submit"> | | | |
| | | | | 65 SUBMIT | | | |
| | | | | 66 </button> | | | |
| | | | | 67 </form> | | | |
| | | | | 68 | | | |
| | | | | 69 Back to Main | | | |
| | | | | 70 | | | |
| | | | | 71 </body> | | | |
| | | | | 72 </html> | | | |

Date 값을 1941-11-03으로 변경한다. Position3 값은 R, Ring3은 9로 변경한다. 내용을 보니 마지막 힌트이다. Roter3값은 직접 대입해서 구한다.

| Request | | | | Response | | | |
|---|-----|-----|--|--|-----|-----|--------|
| Pretty | Raw | Hex | | Pretty | Raw | Hex | Render |
| 1 GET /problems/timemachine/07.php HTTP/1.1 | | | | 52 </p> | | | |
| 2 Host: 192.168.5.166 | | | | 53 <p> | | | |
| 3 User-Agent: Wadeline | | | | 54 <div class="highlight" style="text-align: left; display: inline-block; max-width: 600px;"> | | | |
| 4 Date: 1941-11-03 | | | | 55 <p>Demo: | | | |
| 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 | | | | 56 <p>We have recovered partial information about the final rotor. | | | |
| 6 Accept-Language: en-US,en;q=0.5 | | | | 57 <p>Unfortunately, our agent's identity was compromised – the exact type of ROTORS could not be confirmed. | | | |
| 7 Accept-Encoding: gzip, deflate, br | | | | 58 <p>But before the line went dead, this much was clear: | | | |
| 8 Referer: http://192.168.5.166/main.php | | | | 59 <p>POSITION3 = R | | | |
| 9 Cookie: PHPSESSID=vd2ggj7kbb1s82n10atp6af3k | | | | 60 <p>RING3 = 9 | | | |
| 10 Upgrade-Insecure-Requests: 1 | | | | 61 <p>ROTORS = ??? (unknown) | | | |
| 11 Priority: u=0, i | | | | 62 <p>There will be no more messages. Good luck. | | | |
| 12 | | | | 63 </div> | | | |
| 13 | | | | 64 <form method="post"> | | | |
| 14 | | | | 65 <input type="text" name="flag" placeholder="Enter DECODE" required /> | | | |
| | | | | 66 <button type="submit"> | | | |
| | | | | 67 SUBMIT | | | |
| | | | | 68 </button> | | | |
| | | | | 69 </form> | | | |
| | | | | 70 | | | |
| | | | | 71 Back to Main | | | |
| | | | | 72 | | | |
| | | | | 73 </body> | | | |
| | | | | 74 </html> | | | |

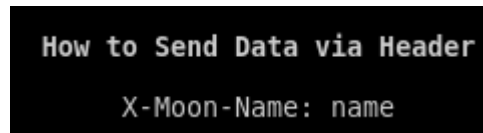
하나씩 대입하다보면 forty weepy weepy라는 단어들이 나온다.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|--|---------------------|-------------|--|--|-----------|---------|--|--|---------|----------|------|--|--------|---------|---------|--|---------|----------|------|--|-----|---------|----------|--|---------|----------|------|--|------|----------|---------|--|-----------|----------------------|--|--|---------------|----------------|--|--|--------------------|--|--|--|-------------------|
| VIEW Ciphertext ▾ | ENCODE DECODE + Enigma machine ▾ | VIEW Plaintext ▾ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| vxxps fucwn chrfd | <table><tr><td>MODEL</td><td colspan="3">Enigma M3 ▾</td></tr><tr><td>REFLECTOR</td><td colspan="3">UKW B ▾</td></tr><tr><td>ROTOR 1</td><td>POSITION</td><td colspan="2">RING</td></tr><tr><td>VIII ▾</td><td>- 5 E +</td><td>- 3 C +</td><td></td></tr><tr><td>ROTOR 2</td><td>POSITION</td><td colspan="2">RING</td></tr><tr><td>V ▾</td><td>- 6 F +</td><td>- 12 L +</td><td></td></tr><tr><td>ROTOR 3</td><td>POSITION</td><td colspan="2">RING</td></tr><tr><td>II ▾</td><td>- 18 R +</td><td>- 9 I +</td><td></td></tr><tr><td>PLUGBOARD</td><td colspan="3">cw nu bd gr eh is ky</td></tr><tr><td>FOREIGN CHARS</td><td colspan="3">Include Ignore</td></tr><tr><td colspan="4">→ Decoded 17 chars</td></tr></table> | MODEL | Enigma M3 ▾ | | | REFLECTOR | UKW B ▾ | | | ROTOR 1 | POSITION | RING | | VIII ▾ | - 5 E + | - 3 C + | | ROTOR 2 | POSITION | RING | | V ▾ | - 6 F + | - 12 L + | | ROTOR 3 | POSITION | RING | | II ▾ | - 18 R + | - 9 I + | | PLUGBOARD | cw nu bd gr eh is ky | | | FOREIGN CHARS | Include Ignore | | | → Decoded 17 chars | | | | forty weepy weepy |
| MODEL | Enigma M3 ▾ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| REFLECTOR | UKW B ▾ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ROTOR 1 | POSITION | RING | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VIII ▾ | - 5 E + | - 3 C + | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ROTOR 2 | POSITION | RING | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| V ▾ | - 6 F + | - 12 L + | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ROTOR 3 | POSITION | RING | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| II ▾ | - 18 R + | - 9 I + | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| PLUGBOARD | cw nu bd gr eh is ky | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FOREIGN CHARS | Include Ignore | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| → Decoded 17 chars | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

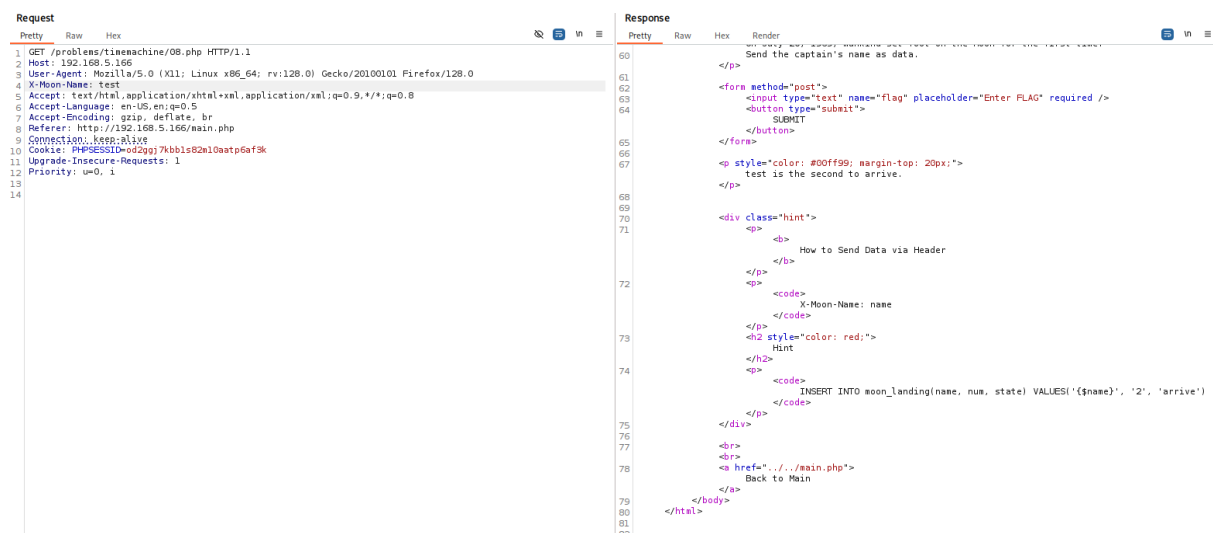
이것을 입력하면 정답.

08. First Moon Landing

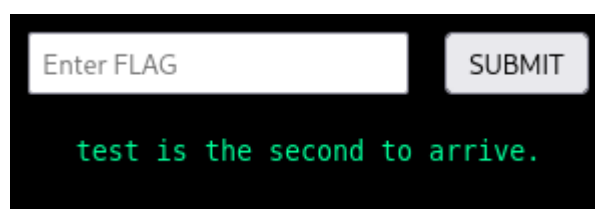
Data를 보내기 위해서는 http header를 이용하여 X-Moon-Name에 name값을 입력하면 된다고 한다.



Burp suite를 통해 header에 X-Moon-Name: test를 입력하면 test is the second to arrive. 라는 문장이 나온다.

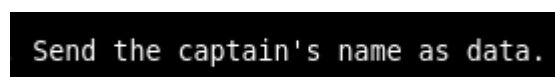


화면에는 다음과 같이 나온다.

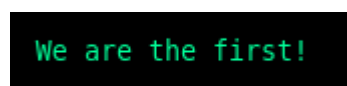


Captain의 이름을 전달해야 하므로 name 값으로 Armstrong으로 해보자.

X-Moon-Name: Armstrong



Armstrong을 입력하면 다음과 같은 문장이 나온다.

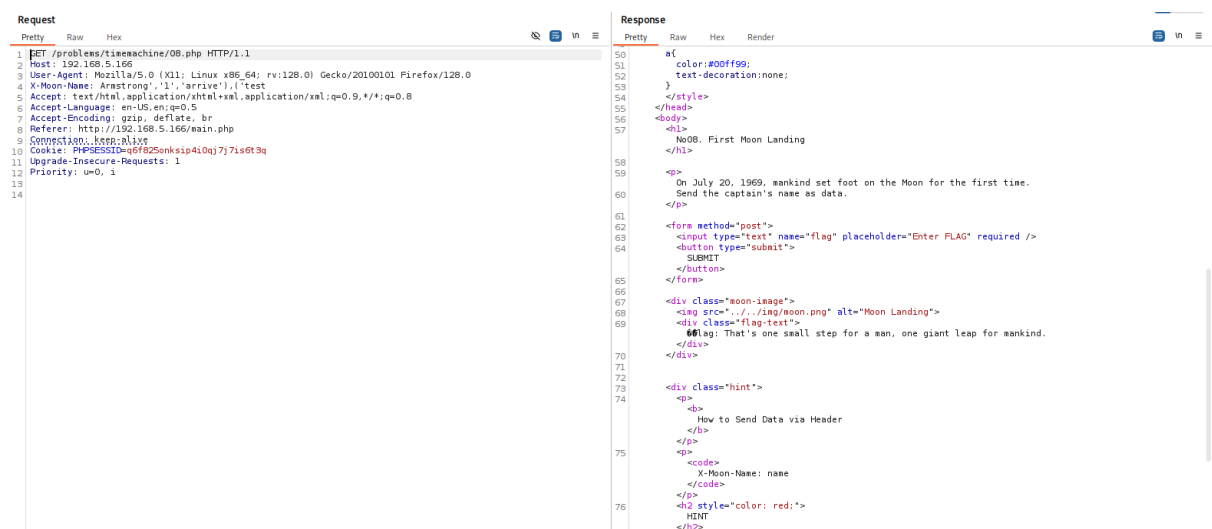


힌트를 보면 insert 구문을 통해 입력된 값에 따라 출력됨을 알 수 있다. Num에 해당하는 값을 1로 수정해야 한다.

Hint

```
INSERT INTO moon_landing(name, num, state) VALUES('${name}', '2', 'arrive')
```

X-Moon-Name: Armstrong,'1','arrive'),('test 를 입력하면 Flag: ~ 가 나온다.



화면에는 다음과 같이 나온다.




이제 Flag 입력칸에 That's one small step for a man, one giant leap for mankind. 를 입력하면 클리어.

09. Cryptocurrency Mining

Nonce 값에 1을 입력했더니 00000으로 시작해야 하고, aaa값이 있어야 한다고 나온다.

| | |
|---------------|--|
| Version | 1 |
| Previous Hash | ef7797e13d3a75526946a3bcf00daec9fc9c9c4d51ddc7cc5df888f74dd434d1 |
| Merkle Hash | 7975edd9e7393c229e744913fe0d0bb86fb4cf46906e2e51152137e20ad15590 |
| Time | 2025-04-30 11:00:00 |
| Bits | 00000fff |
| Nonce | <input type="text" value="1"/> |
| SHA-256 Hash | 3706cefd99e5b949061e261a0f23bf8ba625d25d8985925b3e3d96accf9e4998 |

Mine

 [machine](#)

❌ Invalid hash. It must start with '00000' and contain 'aaa'.

Machine 다운로드 링크를 통해 파이썬 파일을 받고 빈칸에 맞는 값을 입력한다.

```
1 import hashlib
2
3 version = "1" # fill in
4 prev_hash = "ef7797e13d3a75526946a3bcf00daec9fc9c9c4d51ddc7cc5df888f74dd434d1" # fill in
5 merkle = "7975edd9e7393c229e744913fe0d0bb86fb4cf46906e2e51152137e20ad15590" # fill in
6 time = "2025-04-30 11:00:00" # fill in
7 bits = "00000fffffffffffffffffffffffffffffffffffffffffffffffffffff" # fill in
8 target_prefix = "00000" # fill in
9 middle_pattern = "aaa" # fill in
10
11 base = version + prev_hash + merkle + time + bits
12
13 for nonce in range(1000000000): # fill in
14     full = base + str(nonce)
15     h = hashlib.sha256(full.encode()).hexdigest()
16     if h.startswith(target_prefix) and middle_pattern in h:
17         print(f"[✅ FOUND] Nonce: {nonce}")
18         print(f"Hash: {h}")
19
```

PS C:\Users\Administrator\Desktop> & 'c:\Users\Administrator\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\Administrator\.vscode\extensions\c:\Users\Administrator\Desktop\machine.py'

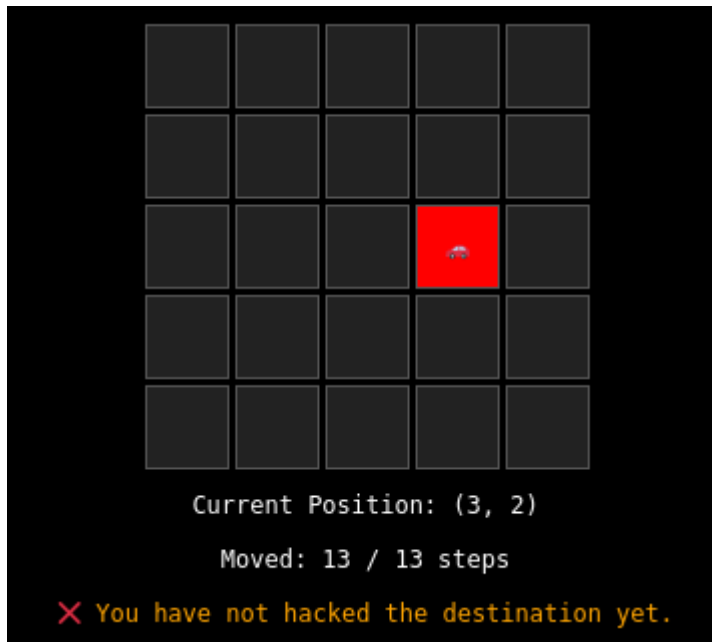
[✅ FOUND] Nonce: 174715282

Hash: 000006d155bdd0537cb5af2d8093785ce95cf2590635e74728adaaac514eb97

나온 값을 입력하고 mine 버튼을 누르면 클리어.

10. Autonomous Driving

Move to destination 버튼을 누르면 자동차 이미지가 이동하는데 아래와 같은 실패 메시지가 나온다.



힌트를 보면 좌표를 해킹해야 한다.

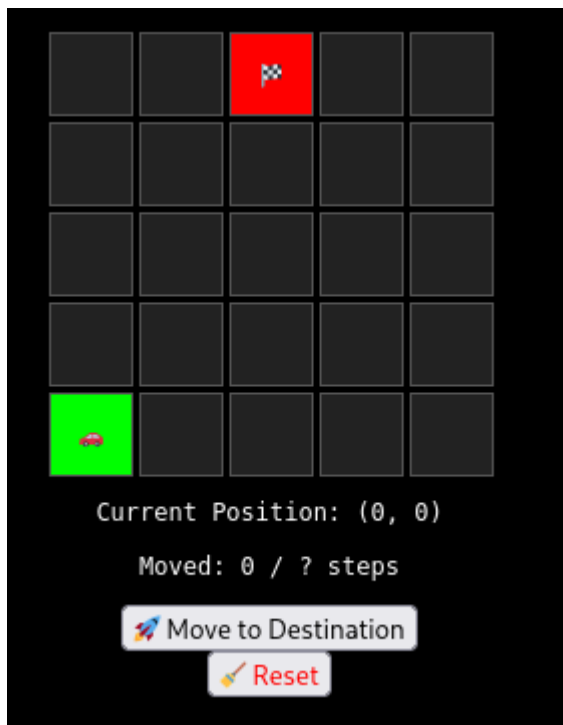
Hint: Submit the hacking coordinates.

Burp suite를 통해 확인하면 destination 값을 확인할 수 있다.

Response

| | Pretty | Raw | Hex | Render |
|----|---|-----|-----|--------|
| 1 | HTTP/1.1 200 OK | | | |
| 2 | Date: Tue, 06 May 2025 02:29:39 GMT | | | |
| 3 | Server: Apache/2.4.62 (Rocky Linux) | | | |
| 4 | X-Powered-By: PHP/8.0.30 | | | |
| 5 | Expires: Thu, 19 Nov 1981 08:52:00 GMT | | | |
| 6 | Cache-Control: no-store, no-cache, must-revalidate | | | |
| 7 | Pragma: no-cache | | | |
| 8 | Keep-Alive: timeout=5, max=93 | | | |
| 9 | Connection: Keep-Alive | | | |
| 10 | Content-Type: application/json | | | |
| 11 | Content-Length: 90 | | | |
| 12 | | | | |
| 13 | <pre>{ "current":{ "x":3, "y":1 }, "destination":{ "x":3, "y":2 }, "count":6, "total":15, "success":false }</pre> | | | |

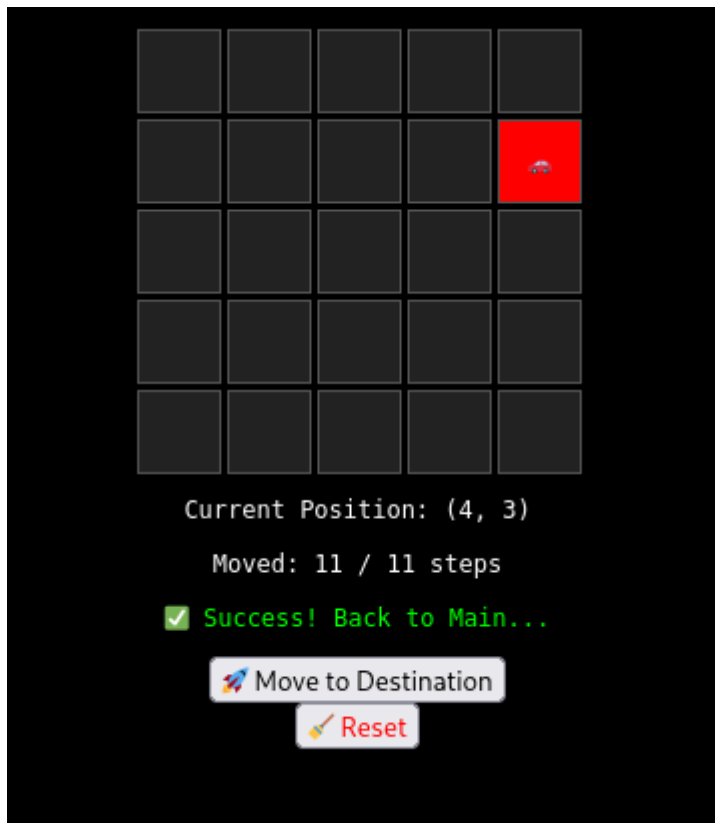
Reset을 한 번 누른 뒤 새로고침을 하여 Burp suite로 intercept할 때 reset=1이 있는데 이를 지우고 x=2&y=4 와 같은 임의값을 넣어본다. 그러면 목적지의 위치가 바뀐다.



(2,4)는 정답이 아님을 알 수 있다.



정답인 목적지 좌표를 찾을 때까지 좌표를 움직이며 확인한다.



(4,3)일 때 클리어.